REMARKS

Claims 1-37 are pending.

Rejections under 35 U.S.C. §103(a)

Independent Claims 1, 13, 25, and 26 and dependent Claims, 2, 4-6, 8-11, 14, 16-18, 20-23, 27, 29-31, and 33-36 stand rejected under 35 U.S.C. §103(a) as being unpatentable over *Nagaoka et al.* (U.S. Patent No. 6,574,656) in view of the Microsoft Press Computer Dictionary (1997).

Dependent Claims 3, 12, 15, 24, 28, and 37 stand rejected under 35 U.S.C. §103(a) as being unpatentable over *Nagaoka et al.* and the Microsoft Press Computer Dictionary and in further view of *Comay et al.* (U.S. Patent No. 6,363,489).

Dependent Claims 7, 19 and 32 stand rejected under 35 U.S.C. §103(a) as being unpatentable over *Nagaoka et al.* and the Microsoft Press Computer Dictionary and in further view of *Skopp et al.* (U.S. Patent No. 6,256,739).

Applicants traverse these rejections for at least the following exemplary reasons, and respectfully request that the rejections be reconsidered and withdrawn.

Nagaoka et al. disclose a system and method for limiting the execution of commands by a shared computer by having an administrator input and establish a list of users that are authorized to have commands executed. This resulting authorization list and corresponding software instructions to verify authorization are user-mode processes in the shared computer.

The Microsoft Press Computer Dictionary is used to define a computer network in the Office Action.

Comay et al. disclose a system and method for detecting unauthorized users in a network by causing an unauthorized user to unknowingly gather and later present certain information that identifies the user as having attempted to gain access when not authorized. The placement and detection of such "mark" data is the result of software instructions that are user-mode processes in the network computer.

Skopp et al. disclose an apparatus and method for determining user identity and controlling access to network resources. User identity is determined via protocol exchanged messages that include information that can be verified by logic running on a proxy controller. The high level protocols, messaging and decisions logic of the proxy are each user-mode processes in their respective devices

Before describing certain exemplary differences between the cited art and the pending claims, important differences between user-mode processes/data and kernel-mode processes/data will be pointed out.

Use-mode processes/data, such as, for example, application code/data, is separate from kernel-mode processes/data and therefore cannot gain access to system data except by calling subsystem-supplied functions, which, in turn, call system services. Kernel-mode processes/data is privileged and includes, for example, the operating system executive code and system data. Thus, for example, a driver or thread running in kernel-mode has access to system memory and hardware.

With this in mind, reference is made to Figs 3 and 4 in the present patent application, wherein prior-art systems are shown as having processes associated with user-mode and kernel-mode. The corresponding detailed description further points out several of the problems that exist in such prior-art systems. These are pointed out herein because the each of the systems in the cited art above appears to clearly fall into the type of system illustrated in Fig. 3, wherein user-mode processes/data is used to determine if a user is allowed to access a shared resource.

To the contrary, the rejected claims are more akin to Figs 5-7, wherein user-mode processes provide information that becomes kernel-mode information for kernel-mode processing and unwanted users/devices/etc. are dealt with more efficiently at the kernel-mode level.

More particularly, independent **Claim 1** is directed to a method for controlling access to a server device by at least one client device that is operatively coupled to the server device through at least one interconnecting network. The method includes causing a user-side portion of network server logic within the server device to selectively specify at least one network from which the user-side portion would accept client device information. The method further includes causing a kernel-side portion of the network server logic to accept the client device information only if the client device information has been provided via the specified network. **Claims 2-12** each depend from independent Claim 1 and recite additional claim limitations.

Nagaoka et al., the Microsoft Press Computer Dictionary, *Comay et al.*, and/or *Skopp et al.*, alone or combined, do not disclose or reasonably suggest such a method. Not one of these references identify or otherwise even come close to realizing that a user-side portion of a network server logic can selectively specify at least one network from which the user-side portion would accept client device information and that a kernel-side portion of the network server logic could be configured to accept the client device information only if the client device information has been provided via the specified network.

Independent **Claim 13** is directed to a computer-readable medium having computer-executable instructions for performing steps that include causing a user-side portion of a network server logic within the server device to selectively specify at least one network from which the user-side portion would accept client device information, and causing a kernel-side portion of the network server logic to accept the client device information only if the client device information has been provided via the specified network. **Claims 14-24** each depend from independent Claim 13 and recite additional claim limitations.

Again, *Nagaoka et al.*, the Microsoft Press Computer Dictionary, *Comay et al.*, and/or *Skopp et al.*, alone or combined, do not disclose or reasonably suggest such steps.

Independent **Claim 25** is directed to a method for establishing per-socket interface listings. The method includes: (a) issuing, by a user-side application, at least one network identifier from which the user-side application would accept client device information; (b) receiving, by a user-side portion of a network server process, the at least one network identifier; (c) issuing, by the user-side portion, the at least one network identifier; and (d) receiving, by a kernel-side portion of a network server process, the at least one network identifier.

These steps and this method are not disclosed or suggested by the user-mode access verification/control techniques taught by *Nagaoka et al.*, the Microsoft Press Computer Dictionary, *Comay et al.*, and/or *Skopp et al.*

Independent **Claim 26** is directed to an apparatus that includes memory and network server logic. The network server logic is operatively coupled to the memory and configurable to support at least one client-server communication session. The network server logic includes a user-side portion that is configured to selectively specify at least one network from which the user-side portion would accept client device information, and a kernel-side portion that is configured to accept the client device information only if the client device information has been provided via the specified network. **Claims 27-37** each depend from independent Claim 26 and recite additional claim limitations.

Nagaoka et al., the Microsoft Press Computer Dictionary, *Comay et al.*, and/or *Skopp et al.* alone or together fail to describe or otherwise suggest such an apparatus.

Consequently, each pending claim is patentable over the cited art.

Conclusion

The pending claims have been placed in condition for allowance and are clearly patentable over the cited art and should therefore be allowed.

Respectfully Submitted,

Date: 2/3/2004

By: ______________________
Thomas A. Jolly
Reg. No. 39,241